

A. Appendix A

We expand and provide more details regarding the different elements of the S2MGen pipeline. To ensure reproducibility of our work, we provide extensive details into all the generation parameters for all the datasets and the training hyperparameters for the experiments.

A.1. Camera Position and Direction

In this section, we show the effects of the parameters p_{focal} and p_{front} on F (Figure 9b) and Θ (Figure 9a) respectively. The equations for this section are described in 3.3. The values of p_{focal} and p_{front} can be tuned to obtain datasets with varying ratios of zoomed-in and face-frontal shots respectively.

A.2. General Experimental Details

For all of the experiments published, we use a U-Net architecture with skip connections, with Kaiming initialization. We use ReLU activations and dropout for regularization. While training on Synthetic data and testing on real datasets, we use 10% of the corresponding real training datasets as the validation set to choose checkpoints for inference.

The loss function used is Cross Entropy loss with inverse frequency weighting. This is important to alleviate the effects of data balance being varied between different datasets. For example, adding clothing or increasing the focal length in the dataset generation pipeline reduces the amount of skin pixels in the training data. This leads to model performance differences being overdependent on changes in data balance. We weight the Cross Entropy loss calculated at every pixel by the factor α based on the implementation by Sushko et al.¹.

$$\alpha = \frac{H * W}{\min(\sum_j^{H*W} I(y_{j,c} = 1), threshold)} \quad (3)$$

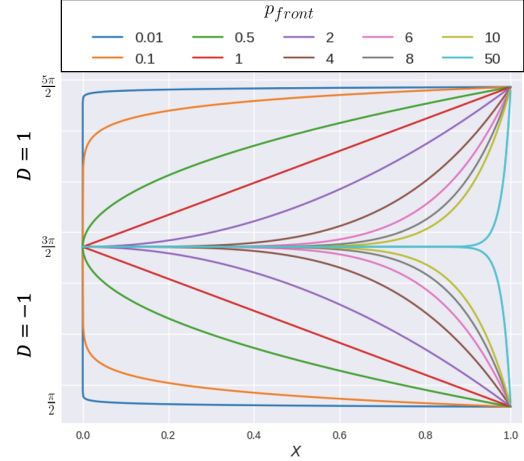
where,

I is an indicator function satisfied if the class c occurs at the pixel j in the ground truth mask y .

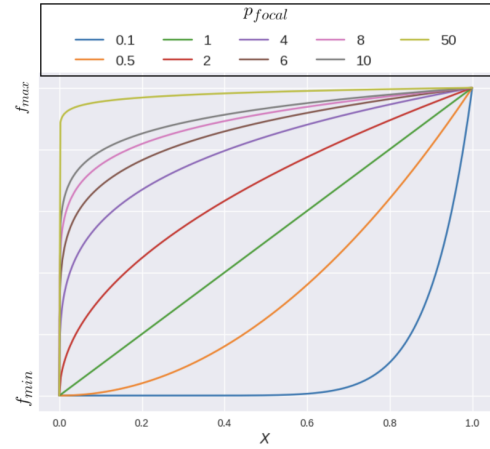
$threshold$ is used to limit α when class occurrence tends towards zero. We set $threshold$ as 10% of the image size.

A.3. Evaluating S2MGen’s Tunable Parameters

We create various versions of the Synthetic Dataset to perform experiments described in Section 4.2. We list the values of the tunable parameters used in Table 4 to create these versions. For the varying focal length experiments, we generate three different datasets (Full Body, Portraits and Faces) by sampling focal length uniformly within the mentioned f_{min} and f_{max} values. We then train on mixed



(a) Varying p_{front} in Equation 1



(b) Varying p_{focal} in Equation 2

Figure 9. **Varying p_{focal} and p_{front} :** Sample transformations of the probability distributions of Θ and F while varying p_{front} and p_{focal} respectively. As we increase p_{front} , the likelihood of face-frontal images increases (Θ is more likely to be closer to $\frac{3\pi}{2}$). As we increase p_{focal} , the likelihood of close-up shots increase (F more likely to be closer to f_{max} than f_{min})

| Exp. Section | Optional Description | Size | $p_{clothing}$ | $p_{background}$ | f_{max} | f_{min} |
|--------------|----------------------|----------|----------------|------------------|-----------|-----------|
| 4.1 | - | variable | 0.8 | 0.8 | 300 | 200 |
| 4.2 | - | 5000 | variable | 0.8 | 200 | 35 |
| 4.2 | - | 5000 | 0.8 | variable | 300 | 200 |
| 4.2 | Full Body | 5000 | 0.8 | 0.8 | 200 | 35 |
| 4.2 | Portraits | 5000 | 0.8 | 0.8 | 300 | 200 |
| 4.2 | Faces | 5000 | 0.8 | 0.8 | 310 | 300 |
| 4.3 - 4.7 | - | 8000 | 0.8 | 0.8 | 300 | 200 |

Table 4. **Parameter Values Used to Create Synthetic Dataset Versions:** We list all the parameter values used to create dataset versions used for experiments in Section 4.

versions of these datasets by varying the sampling ratios between them. However, this can also be approximately achieved one-shot by solving for p_{focal} in equation 2 by

¹Sushko, Vadim, et al. "You only need adversarial supervision for semantic image synthesis." arXiv preprint arXiv:2012.04781 (2020).



Figure 10. **More Qualitative Results:** We show some examples of performance gain we observe when doing cross-dataset and in-dataset inference on the ECU, SFA, HGR, and the Abdomen dataset with and w/o pretrained model on synthetic data generated from S2MGen.

substituting X and F with the required ratio and transition focal length respectively.

For all of the finetuning experiments in the following sections, we pretrain the model on synthetic data generated

| Experiment | IoU | Acc | F1-score |
|---|--------|--------|----------|
| ECU Training + SimCLR | 0.8156 | 0.96 | 0.8925 |
| ECU Training + Synthetic Pretraining + SimCLR | 0.8328 | 0.9649 | 0.9034 |

Table 5. Effect of SimCLR [9] pretraining on ECU performance. We see that SimCLR initialization for the U-Net backbone improves performance for both with and without synthetic pretraining.

using the paramters described in the last row of 4 and fine-tune for 10 epochs with a learning rate of 10^{-6} .

A.4. Real to Synthetic Domain Gap

In this section, we expand upon the experiment details of section 4.5.

A.4.1 Supervised Domain Adaptation

Finetuning: The finetuning details are described in A.3.

Balanced Gradient Contribution: Balanced Gradient Contribution (BGC) is a method of regularizing the target domain t (real), using the source domain s (synthetic), that was previously explored as a method of adaptation in Ros et al. [59]. We implement BGC on every mini-batch update using the following equation:

$$\ell_{BGC}(X, Y) = \ell(X_t, Y_t) + \lambda \ell(X_s, Y_s)$$

where λ is chosen as 0.9 and ℓ is Cross Entropy loss with inverse frequency weighting as in Eq. 3.

A.4.2 Unsupervised Domain Adaptation

Fourier Domain Adaptation: FDA [78] is a domain alignment process, where the low frequency spectrum of source images are replaced with that of the target images. This allows the model to learn useful higher order information and remain unaffected by the misalignment of low-level image statistics. The FDA experiments are done with single-scale, with $\beta = 0.01$ and without entropy minimization.

DANN: DANN [17] uses an adversarial domain alignment approach. The network is trained to learn domain-invariant intermediate features with the use of a domain classifier/discriminator block. For the DANN experiments, we redesign the discriminator as a PatchGAN discriminator structure trained with weight annealing in the initial epochs.

PixMatch: PixMatch [41] uses psuedolabels to regularize the model as a method of domain adaptation. During training, by constraining the model to predict consistent labels on the target domain with and without perturbations, the network learns to adapt to target domains. For the PixMatch experiments, we enforce only the augmentation consistency with a weight of 0.1.

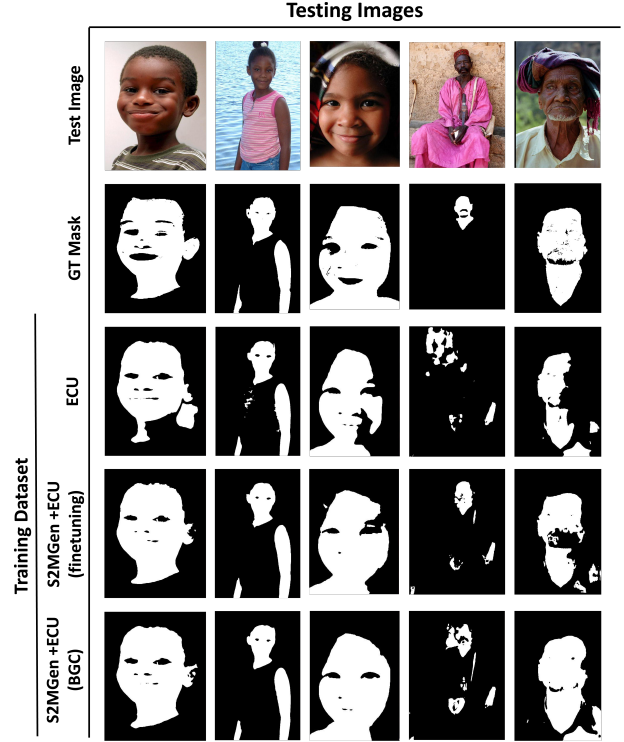


Figure 11. **Skin Tone Qualitative Examples:** We show examples of improvement in segmentation performance on images belonging to Fitzpatrick skintone scale 6 by the injection of S2MGen into the training pipeline

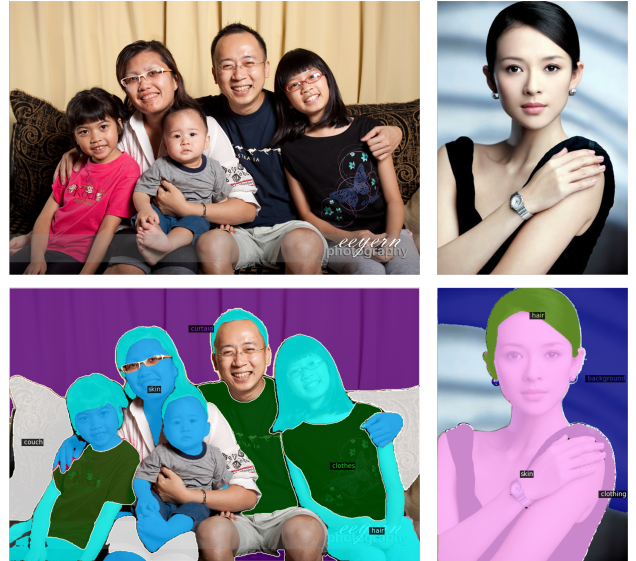


Figure 12. **Foundational Models:** Results of Skin Segmentation with open source Foundational Segmentation Model OV-Seg

A.5. Effects of Unsupervised Pretraining

We experiment with unsupervised pretraining in 5 using SimCLR. We pretrain the U-Net encoder for 200 epochs on 50k images from the OpenImagesV6 dataset [34]. We see general improvement in model performance through SimCLR initialized weights, however subsequently pretraining the model with synthetic data shows a considerable improvement in performance. This shows that performance gain by synthetic data pretraining is additive and can be used in addition with other augmentations/improvements.

A.6. Foundational Models

Recently foundational models such as SAM [29] have shown incredible progress in promptable image segmentation. Extensions of SAM using CLIP [52] to allow for text prompts have been explored in OV-Seg [37], LangSAM [18] etc. We hoped to compare the performance of supervised synthetic data with unsupervised data annotated using foundational segmentation models. But as shown in Fig. 12, initial experiments for CLIP based prompting show insufficient results for skin segmentation. The model often misses skin regions and instead picks up hair (*left example*) and clothing (*right example*) as skin.

A.7. Qualitative Analysis Continued

In continuation with section 4.4, we show more qualitative examples in Fig. 10 of adding S2MGen data into the training pipeline. In each subfigure, we show the results on a randomly selected image from each of the test datasets, on three models - the third row corresponds to a model trained only on synthetic data, the fourth row are the results of a model trained only on the specific real training set and the last row corresponds to the results of a model pretrained on synthetic data and finetuned on the real training set. As observed in the section 4.4, we see that S2MGen performs reasonably well and the performance is improved by adding real data, however biased the real datasets are. S2MGen also improves the performance of the real datasets, especially in cross-dataset performance.

A.8. Skintone Diversity - Qualitative Analysis

In this section, we show qualitative results for section 4.7. We sample images from the ECU dataset that belong to the Fitzpatrick 6 skintone group using the skintone labels from Xu et. al [76]. We show performance improvement by the injection of synthetic data, both by finetuning and BGC.